

Application No. : 09/808,469
Filed : March 14, 2001

~~APPENDIX III~~

~~Copyright © 2000-2001 ARC International plc. All rights reserved.~~

Application No. : 09/808,469
Filed : March 14, 2001

```
#
# ----- Confidential Information
# ----- Limited Distribution to Authorized Persons Only
# ----- Created 2000 and Protected as an Unpublished Work
5 # ----- Under the U.S. Copyright act of 1976.
# ----- Copyright © 2000-2001 ARC CORES LTD
# ----- All Rights Reserved
#
#
10 # SCCS release : %M% %I% %G%
#
# Description : Script to analyse an ARC assembler file and
# print frequency of usage stats for various
# proposed ARC instruction formats
#
15 #
#
#--
=====
=====
20 #
BEGIN {
-out = "e"
-reg = "%r(0|1|2|3|13|14|15|16),"
-reg = "%r(0|1|2|3|13|14|15|16) ([^0-9]|$)"
25 -regh = "%(r[0-9]+|sp|fp|gp|blink) ([^0-9]|$)"
-reg01 = "%r(0|1) ([^0-9]|$)"
-reg23 = "%r(2|3) ([^0-9]|$)"
-reg1316 = "%r(13|14|15|16) ([^0-9]|$)"
-pete = 0
30 -printf "" >out
+

function nxt() {
-print $0 >>out
35 -next
+
function nxte() {
-print "e" $0 >>out
-nxt
40 +

$1 == "bl" {
-bl++
45 -if ($2 ~ /__prolog_.*/) {
--push++
-nxt()
} else {
--calls[$2]++
-nxte()
50 +
+
$1 == "b" {
-b++
55 -if ($2 ~ /__epilog_.*/) {
--pop++
-nxt()
} else {
-nxte()
+
60 +
$1 == "beq" || $1 == "bne" {
-if ($2 !~ /__epilog_.*/) {
--beq++
-nxte()
65 -} else {
-nxt()
+
+
}
```

Application No. : 09/808,469
Filed : March 14, 2001

```
5  +
   $1 == "bgt" || $1 == "ble" || $1 == "bge" || $1 == "blt" {
   if ($2 != /__epilog_.*/) {
       bgt++
       nste()
     } else {
       nst()
     }
10 +
   $1 == "bhi" || $1 == "blo" || $1 == "bhs" || $1 == "blo" {
   if ($2 != /__epilog_.*/) {
       bhi++
       nst()
     } else {
15     nst()
     }
   +
   $1 == "bpl" || $1 == "bmi" {
   if ($2 != /__epilog_.*/) {
20     bpl++
       nst()
     } else {
       nst()
     }
25 +
   $1 == "jeq" || $1 == "jne" {
   if ($2 == "blink") {
       beq++
       nste()
30   }
       nst()
   +
   $1 == "jgt" || $1 == "jle" || $1 == "jge" || $1 == "jlt" {
   if ($2 == "blink") {
35     bgt++
       nste()
     }
       nst()
   +
   $1 == "j" {
   if ($2 == "blink") {
40     jblink++
       nste()
     }
45 if ($2 == reg) {
       jr++
       nste()
     }
       nst()
50 +
   $1 == "jl" {
   if ($2 == reg) {
       jlr++
       nste()
55   }
       nst()
   +
   $1 == "ld" {
   if ($2 == reg) {
60     ld++
   if ($3 == "[%fp,") {
     # ldffa[$4]++
     ldffp++
   if (($4+0) >= -32 && ($4+0) <= -4) {
65   ldffp32++
       nste()
     }
```

Application No. : 09/808,469
 Filed : March 14, 2001

```

  —  nxt()
  —  }
  —  if ($3 == "[%sp,") {
5  # — ldspa[$4]++
  —  ldsp++
  —  nxt()
  —  }
  —  if ($3 == "[%gp,") {
10 —  ldgp++
  —  nxtc()
  —  }
  —  if ($3 ~ reg) {
  # — ldra[$4]++
  —  ldr++
15 —  if ($3 ~ /\]/ || ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 64)) {
  —  ldr64++
  —  nxtc()
  —  }
  —  if (pete) {
20 —  if ($3 ~ /\]/ || ($3 ~ reg01 && ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 128))) {
  —  ldr64p++
  —  nxtc()
  —  }
  —  if ($3 ~ /\]/ || ($3 ~ reg23 && ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 64))) {
25 —  ldr64p++
  —  nxtc()
  —  }
  —  if ($3 ~ /\]/ || ($3 ~ reg1316 && ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 32))) {
30 —  ldr64p++
  —  nxtc()
  —  }
  —  if ($4 ~ reg) {
35 —  ldabc++
  —  nxtc()
  —  }
  —  nxtc()
  —  }
40 —  nxt()
  —  }
  $1 == "ldw" {
  if ($2 ~ reg) {
45 —  ldw++
  —  if ($3 == "[%fp,") {
  —  ldwfp++
  —  if (($4+0) >= 32 && ($4+0) <= 4) {
  —  ldwfp32++
50 —  nxtc()
  —  }
  —  nxtc()
  —  }
  —  if ($3 == "[%sp,") {
55 —  ldwsp++
  —  nxtc()
  —  }
  —  if ($3 == "[%gp,") {
  —  ldwgp++
60 —  nxtc()
  —  }
  —  if ($3 ~ reg) {
  —  ldwr++
  —  if ($3 ~ /\]/ || ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 32)) {
65 —  ldwr32++
  —  nxtc()
  —  }

```

Application No. : 09/808,469
Filed : March 14, 2001

```

if ($4 ~ reg) {
ldwabe++
nxt()
}
5 nxt()
}
nxt()
}
10 $1 == "ldb" {
if ($2 ~ reg) {
ldb++
if ($3 == "[%fp,") {
ldbf++
15 if (($4+0) >= 32 && ($4+0) <= 4) {
ldbf32++
nxt()
}
nxt()
20 }
if ($3 == "[%sp,") {
ldbsp++
nxt()
}
25 if ($3 == "[%gp,") {
ldgpp++
nxt()
}
if ($3 ~ reg) {
30 ldbr++
if ($3 ~ /\]/ || ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 16)) {
ldbr16++
nxtc()
}
35 if ($4 ~ reg) {
ldbabe++
nxt()
}
nxt()
40 }
nxt()
}
stblink, \[%sp, 4\]/ {
45 stblink++
nxtc()
}
$1 == "st" {
if ($2 ~ reg) {
50 st++
if ($3 == "[%fp,") {
# stfpa[$4]++
stfp++
if (($4+0) >= 32 && ($4+0) <= 4) {
55 stfp32++
nxtc()
}
nxt()
}
60 if ($3 == "[%sp,") {
# stopa[$4]++
stsp++
nxt()
}
65 if ($3 == "[%gp,") {
stgpp++
nxt()

```

Application No. : 09/808,469
Filed : March 14, 2001

```

    }
    if ($3 ~ reg) {
#   strc[$4]++
    str++
5   if ($3 ~ /\]/ || ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 64)) {
    str64++
    nextc()
    }
    next()
10  }
    }
    next()
    +
    $1 == "stw" {
15  if ($2 ~ reg) {
    stw++
    if ($3 == "[%fp,") {
#   stwfpa[$4]++
    stwfp++
20  if (($4+0) >= 32 && ($4+0) <= 4) {
    stwfp32++
    next()
    }
    next()
25  }
    if ($3 == "[%sp,") {
#   stwsppa[$4]++
    stwspp++
    next()
30  }
    if ($3 == "[%gp,") {
    stwgp++
    next()
    }
35  if ($3 ~ reg) {
#   stwra[$4]++
    stwr++
    if ($3 ~ /\]/ || ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 16)) {
    stwr16++
40  nextc()
    }
    next()
    }
    next()
45  }
    +
    $1 == "stb" {
    if ($2 ~ reg) {
    stb++
50  if ($3 == "[%fp,") {
#   stbfpa[$4]++
    stbfp++
    if (($4+0) >= 32 && ($4+0) <= 4) {
    stbfp32++
55  next()
    }
    next()
    }
    if ($3 == "[%sp,") {
60  #   stbsppa[$4]++
    stbspp++
    next()
    }
    if ($3 == "[%gp,") {
65  stbgp++
    next()
    }
    }

```

Application No. : 09/808,469
 Filed : March 14, 2001

```

if ($3 ~ reg) {
# stbra[$4]++
stbr++
if ($3 ~ /\]/ || ($4 ~ /^[0-9]/ && ($4+0) >= 0 && ($4+0) < 8)) {
5 stbr8++
nxtc()
}
nxt()
}
10 nxt()
}
$1 == "mov.f" {
if ($2 == "0," && $3 ~ reg) {
15 movf0r++
nxtc()
} if ($2 == "0," && $3 ~ regh) {
movf0h++
nxtc()
20 }
nxt()
}
$1 == "mov" {
if ($3 ~ /^[^?]{0-9}/) {
25 movi++
movia[$3]++
if ($2 ~ reg) {
if ($3 >= 0 && $3 < 64) {
movi64++
30 nxtc()
}
if (pete) {
if ($2 ~ reg01 && $3 >= 0 && $3 < 128) {
movi64p++
35 nxtc()
}
if ($2 ~ reg23 && $3 >= 0 && $3 < 64) {
movi64p++
nxtc()
40 }
if ($2 ~ reg1316 && $3 >= 0 && $3 < 32) {
movi64p++
nxtc()
}
45 }
if ($3 < 256 || $3 > 255) {
ldrpe++
nxtc()
}
50 }
nxt()
}
if ($3 ~ reg) {
if ($2 ~ reg) {
55 movr++
nxtc()
}
}
if ($2 ~ reg) {
if ($3 ~ regh) {
60 movrh++
nxtc()
}
}
65 if ($2 ~ regh) {
if ($3 ~ reg) {
movhr++

```

Application No. : 09/808,469
 Filed : March 14, 2001

```

— npte()
— }
— }
5 — if ($3 != /\%/ && $2 == reg) {
— ldrpe++
— npte()
— }
— npt()
— }
10 $1 == "add" {
— if ($2 == $3 || $2 == ($3 " " || $2 == ($4 " "))) {
— if ($4 == /\^?[0-9]/) {
— addi++
# addia[$4]++
15 — if ($3 == reg) {
— if ($4 >= 32 && $4 < 0) {
— subi32++
— npte()
— }
20 — if ($4 >= 0 && $4 < 32) {
— addi32++
— npte()
— }
— }
25 — }
— if ($2 == reg && $3 == reg && $4 == reg) {
— addaab++
— npte()
— }
30 — if ($2 == reg && $3 == reg && $4 == regh) {
— addrhh++
— npte()
— }
35 — if ($2 == reg && $3 == regh && $4 == reg) {
— addrhh++
— npte()
— }
— }
40 — if ($4 == /\^?[0-9]/) {
— if ($2 == reg) {
— if ($3 == reg) {
— if ($4 >= 8 && $4 < 0) {
— subabi8++
— npte()
45 — }
— if ($4 >= 1 && $4 <= 8) {
— addabi8++
— npte()
— }
50 — }
— if ($3 == "%fp") {
— if ($4 >= 32 && $4 < 0) {
— addfpi32++
— npte()
55 — }
— }
— if ($3 == /\%r([12][0-9])/ && $4 >= 512 && $4 < 512) {
— addrpe++
— npte()
60 — }
— }
— npt()
— }
65 — if ($2 == reg && $3 == reg && $4 == reg) {
— addrhh++
— npte()
— }

```


Application No. : 09/808,469
 Filed : March 14, 2001

```

+
$1=="sub" {
  if ($4 ~ /^-?[0-9]/) {
5    sub1++
    if ($2 == $3) {
      # subia[$4]++
      if ($3 ~ reg) {
        if ($4 >= 32 && $4 < 0) {
10          addi32++
          npte()
        }
        if ($4 >= 0 && $4 < 32) {
          sub132++
          npte()
15        }
      }
    }
    if ($2 ~ reg) {
      if ($3 ~ reg) {
20        if ($4 >= 8 && $4 < 0) {
          addabi8++
          npte()
        }
        if ($4 >= 1 && $4 < 8) {
25          subabi8++
          npte()
        }
      }
    }
30    npt()
  }
  if ($2 == $3 && $2 == ($4 ",")) {
    if ($2 ~ reg && $3 ~ reg && $4 ~ reg) {
      subaaa++
35      npte()
    }
    if ($2 ~ regh && $3 ~ regh && $4 ~ regh) {
      subhhh++
      npte()
40    }
  }
  if ($2 ~ reg) {
    subr++
    if ($2 == $3) {
45      if ($2 ~ reg && $3 ~ reg && $4 ~ reg) {
        subaab++
        npte()
      }
      if ($2 ~ reg && $3 ~ reg && $4 ~ regh) {
50        subrrh++
        npte()
      }
      if ($2 ~ reg && $3 ~ regh && $4 ~ reg) {
        subrrh++
55        npte()
      }
    }
    if ($3 ~ reg && $4 ~ reg) {
      subrrr++
60      npte()
    }
  }
  npt()
}
+
65 $1=="sub.f" {
  if ($2 == "0,") {
    if ($3 ~ reg && $4 ~ reg) {

```

Application No. : 09/808,469
 Filed : March 14, 2001

```

empr++
nxtc()
}
5 if ($4 ~ /^-?[0-9]/) {
empr++
# empia[$4]++
if ($3 ~ reg) {
if ($4 >= 0 && $4 < 64) {
empi64++
10 nxtc()
}
if (pete) {
if ($3 ~ reg01 && $4 >= 0 && $4 < 128) {
empi64p++
15 nxtc()
}
if ($3 ~ reg23 && $4 >= 0 && $4 < 64) {
empi64p++
nxtc()
20 }
if ($3 ~ reg1316 && $4 >= 0 && $4 < 32) {
empi64p++
nxtc()
}
25 }
nxtc()
}
if ($3 ~ reg) {
30 if ($4 ~ regh) {
emprh++
nxtc()
}
}
35 if ($3 ~ regh) {
if ($4 ~ reg) {
emphr++
nxtc()
}
40 }
nxtc()
}
45 $1 == "sub.ne" {
if ($2 == $3 && $2 == ($4 ",")) {
if ($4 ~ reg && $2 ~ reg && $3 ~ reg) {
subneaaaa++
nxtc()
}
50 }
nxtc()
}
$1 == "sub.eq" {
55 if ($2 == $3 && $2 == ($4 ",")) {
if ($4 ~ reg && $2 ~ reg && $3 ~ reg) {
subeqaaa++
nxtc()
}
}
60 nxtc()
}
$1 == "asl" {
if ($4 ~ /^-?[0-9]/) {
asli++
65 if ($2 == $3) {
# aslia[$4]++
if ($3 ~ reg) {

```

Application No. : 09/808,469
 Filed : March 14, 2001

```

  if ($4 >= 1 && $4 <= 8) {
    asli8++
  }
  if ($4 >= 1 && $4 < 32) {
5    asli32++
  }
  npte()
}
}
10 if ($2 ~ reg) {
  if ($3 ~ reg && $4 >= 2 && $4 < 3) {
    aslab2++
    npte()
  }
15 }
  npt()
}
  if ($4 ~ reg && $2 ~ reg && $3 ~ reg) {
20    aslaab++
    npte()
  }
  if ($2 ~ reg && $3 ~ reg && $4 !~ reg) {
    aslab1++
    npte()
25 }
  +
  $1 == "asr" {
    if ($4 ~ /^?[0-9]/) {
30      asri++
      if ($2 == $3) {
        # asria[$4]++
        if ($3 ~ reg) {
          if ($4 >= 1 && $4 <= 8) {
            asri8++
35          }
          if ($4 >= 1 && $4 < 32) {
            asri32++
          }
          npte()
40        }
      }
      if ($2 ~ reg) {
        if ($3 ~ reg && $4 >= 2 && $4 < 3) {
45          asrab2++
          npte()
        }
      }
      npt()
    }
50    if ($4 ~ reg && $2 ~ reg && $3 ~ reg) {
      asraab++
      npte()
    }
    if ($2 ~ reg && $3 ~ reg && $4 !~ reg) {
55      asrab1++
      npte()
    }
    +
    $1 == "lsl" {
60      if ($4 ~ /^?[0-9]/) {
        lori++
        if ($2 == $3) {
          # lsria[$4]++
          if ($3 ~ reg) {
65            if ($4 >= 1 && $4 <= 8) {
              lori8++
            }
          }
        }
      }
    }
  }

```

Application No. : 09/808,469
Filed : March 14, 2001

```

5  if ($4 >= 1 && $4 < 32) {
    lsr i32++
  }
  npte()
}

10 if ($2 ~ reg) {
  if ($3 ~ reg && $4 >= 2 && $4 < 3) {
    lsr ab2++
    npte()
  }
  npt()
}

15 if ($4 ~ reg && $2 ~ reg && $3 ~ reg) {
  lsraab++
  npte()
}

20 if ($2 ~ reg && $3 ~ reg && $4 !~ reg) {
  lsra b1++
  npte()
}
+
25 $1 == "mul64" {
  if ($2 == "0,") {
    if ($4 ~ /^?[0-9]/) {
      mul i++
      # mul ia[$4]++
      if ($3 ~ reg) {
30    if ($4 >= 0 && $4 < 32) {
        mul i32++
        npte()
      }
    }
  }
  if ($3 ~ reg && $4 ~ reg) {
    mul0ab++
    npte()
  }
40  +
  npt()
  +
  $1 == "and.f" {
    if ($2 == "0,") {
45    if ($4 ~ /^?[0-9]/) {
      andfi++
      # andfia[$4]++
      if ($3 ~ reg) {
        if ($4 >= 0 && $4 < 32) {
50    andfi32++
        npte()
      }
    }
  }
  if ($3 ~ reg && $4 ~ reg) {
    andfab++
    npte()
  }
60  +
  npt()
  +
  $1 == "and" {
    if ($2 == $3 || $2 == ($3 " " || $2 == ($4 " "))) {
65    if ($4 ~ /^?[0-9]/) {
      andi++
      # andia[$4]++
      if ($3 ~ reg) {

```

Application No. : 09/808,469
 Filed : March 14, 2001

```

if ($4 >= 0 && $4 < 32) {
andi32++
nxtc()
}
5 }
}
if ($2 ~ reg && $3 ~ reg && $4 ~ reg) {
andaab++
nxtc()
10 }
}
if ($2 ~ reg && $3 ~ reg && $4 ~ reg) {
andrrr++
nxtc()
15 }
}
$1 == "extb" {
if ($2 == ($3 ",")) {
if ($2 ~ reg && $3 ~ reg) {
20 extbr++
nxtc()
}
}
nxtc()
25 }
$1 == "extw" {
if ($2 == ($3 ",")) {
if ($2 ~ reg && $3 ~ reg) {
extwr++
30 nxtc()
}
}
nxtc()
}
35 $1 == "sexb" {
if ($2 == ($3 ",")) {
if ($2 ~ reg && $3 ~ reg) {
sexbr++
nxtc()
40 }
}
nxtc()
}
$1 == "sexw" {
45 if ($2 == ($3 ",")) {
if ($2 ~ reg && $3 ~ reg) {
sexwr++
nxtc()
}
50 }
nxtc()
}
($2 == $3 || $2 == ($3 ",") || $2 == ($4 ",")) {
55 if ($1 == "add" || $1 == "sub" || $1 == "and" || $1 == "or" || $1 == "xor" || $1 ==
"asl" || $1 == "asr" || $1 == "lsl") {
if ($2 ~ reg) {
if ($2 == $3) {
if ($4 ~ reg) {
opaab[$1]++
60 nxtc()
}
} else {
if ($3 ~ reg && $2 == ($4 ",")) {
opaab[$1]++
65 nxtc()
}
}

```

```

  }
  }
  +
5  +
  -nxt()
  # print $0
  +
10  END {
  if (1) {
  -OFS = "\t"
  # print "\nopaab"
  for (i in opaab) {
15  -if (i == "add" || i == "sub" || i == "and" || i == "or" || i == "xor" || i == "asl"
  || i == "asr" || i == "lsl") {
  -print i, opaab[i], int(opaab[i]*1000/NR)/10
  -}
  -}
20  # print "\nldfpa"
  # for (i in ldfpa) print i, ldfpa[i]
  # print "\nstfpa"
  # for (i in stfpa) print i, stfpa[i]
  # print "\nlldr0a"
25  # for (i in ldr0a) print i, ldr0a[i]
  # print "\nmovia"
  # for (i in movia) print i, movia[i]
  # print "\naddia"
  # for (i in addia) print i, addia[i]
30  # print "\nsubia"
  # for (i in subia) print i, subia[i]
  # print "\ncmpia"
  # for (i in cmpia) print i, cmpia[i]

35  -for (i in calls) {
  # print i, calls[i]
  -if (calls[i] > 1) {
  -calls2 += (calls[i]-2)
  -}
40  -callsall += calls[i]
  -}
  # print "callsall", callsall, int(callsall*1000/NR)/10
  # print "calls2", calls2, int(calls2*1000/NR)/10

45  # bl = calls2
  -bl = bl - push
  -b = b - pop
  -print "bl", bl, int(bl*1000/NR)/10
  # print "push", push, int(push*1000/NR)/10

50  -print "b", b, int(b*1000/NR)/10
  # print "pop", pop, int(pop*1000/NR)/10
  -print "beq", beq, int(beq*1000/NR)/10
  -print "bgt", bgt, int(bgt*1000/NR)/10
55  -print "bhi", bhi, int(bhi*1000/NR)/10
  -print "bpl", bpl, int(bpl*1000/NR)/10

  -print "stblink", stblink, int(stblink*1000/NR)/10
  -print "jblink", jblink, int(jblink*1000/NR)/10
60  -print "jrr", jrr, int(jrr*1000/NR)/10
  -print "jlr", jlr, int(jlr*1000/NR)/10

  -print "movr", movr, int(movr*1000/NR)/10
  -print "movf0r", movf0r, int(movf0r*1000/NR)/10
65  -print "movf0h", movf0h, int(movf0h*1000/NR)/10
  -print "movrh", movrh, int(movrh*1000/NR)/10
  -print "movhr", movhr, int(movhr*1000/NR)/10

```

Application No. : 09/808,469
Filed : March 14, 2001

```
5  print "emprh", emprh, int(emprh*1000/NR)/10  
print "emphr", emphr, int(emphr*1000/NR)/10  
print "empr", empr, int(empr*1000/NR)/10  
  
10 print "empi64", empi64, int(empi64*1000/NR)/10  
print "empi64p", empi64p, int(empi64p*1000/NR)/10  
print "movi64", movi64, int(movi64*1000/NR)/10  
print "movi64p", movi64p, int(movi64p*1000/NR)/10  
  
15 print "addi32", addi32, int(addi32*1000/NR)/10  
print "subi32", subi32, int(subi32*1000/NR)/10  
  
print "addabi8", addabi8, int(addabi8*1000/NR)/10  
print "subabi8", subabi8, int(subabi8*1000/NR)/10  
  
print "subneaaa", subneaaa, int(subneaaa*1000/NR)/10  
print "subeqaaa", subeqaaa, int(subeqaaa*1000/NR)/10  
  
20 print "subhhh", subhhh, int(subhhh*1000/NR)/10  
print "subaaa", subaaa, int(subaaa*1000/NR)/10  
print "subaab", subaab, int(subaab*1000/NR)/10  
print "subrrr", subrrr, int(subrrr*1000/NR)/10  
  
25 print "addaab", addaab, int(addaab*1000/NR)/10  
print "addrrr", addrrr, int(addrrr*1000/NR)/10  
print "addrrh", addrrh, int(addrrh*1000/NR)/10  
  
print "asli8", asli8, int(asli8*1000/NR)/10  
30 # print "asli32", asli32, int(asli32*1000/NR)/10  
print "aslab1", aslab1, int(aslab1*1000/NR)/10  
print "aslab2", aslab2, int(aslab2*1000/NR)/10  
print "aslaab", aslaab, int(aslaab*1000/NR)/10  
  
print "asri8", asri8, int(asri8*1000/NR)/10  
35 # print "asri32", asri32, int(asri32*1000/NR)/10  
print "asrab1", asrab1, int(asrab1*1000/NR)/10  
print "asrab2", asrab2, int(asrab2*1000/NR)/10  
print "asraab", asraab, int(asraab*1000/NR)/10  
  
40 print "lsri8", lsri8, int(lsri8*1000/NR)/10  
# print "lsri32", lsri32, int(lsri32*1000/NR)/10  
print "lsrab1", lsrab1, int(lsrab1*1000/NR)/10  
print "lsrab2", lsrab2, int(lsrab2*1000/NR)/10  
  
45 print "lsraab", lsraab, int(lsraab*1000/NR)/10  
  
print "andi32", andi32, int(andi32*1000/NR)/10  
print "andfi32", andfi32, int(andfi32*1000/NR)/10  
print "andaab", andaab, int(andaab*1000/NR)/10  
  
50 print "andfab", andfab, int(andfab*1000/NR)/10  
  
print "mul0ab", mul0ab, int(mul0ab*1000/NR)/10  
print "muli32", muli32, int(muli32*1000/NR)/10  
  
print "ldabe", ldabe, int(ldabe*1000/NR)/10  
55 print "ldbabe", ldbabe, int(ldbabe*1000/NR)/10  
print "ldwabe", ldwabe, int(ldwabe*1000/NR)/10  
print "ldr64", ldr64, int(ldr64*1000/NR)/10  
print "ldr64p", ldr64p, int(ldr64p*1000/NR)/10  
  
60 print "ldwr32", ldwr32, int(ldwr32*1000/NR)/10  
print "ldbr16", ldbr16, int(ldbr16*1000/NR)/10  
print "str64", str64, int(str64*1000/NR)/10  
print "stbr8", stbr8, int(stbr8*1000/NR)/10  
  
print "stwr16", stwr16, int(stwr16*1000/NR)/10  
  
65 print "ldrpe", ldrpe, int(ldrpe*1000/NR)/10  
print "addrpe", addrpe, int(addrpe*1000/NR)/10
```

Application No. : 09/808,469
Filed : March 14, 2001

```
-print "ldfp32", ldfp32, int(ldfp32*1000/NR)/10
-print "stfp32", stfp32, int(stfp32*1000/NR)/10
-print "addfpi32", addfpi32, int(addfpi32*1000/NR)/10

5 -print "ldgp", ldgp, int(ldgp*1000/NR)/10
-print "stgp", stgp, int(stgp*1000/NR)/10

-print "extbr", extbr, int(extbr*1000/NR)/10
-print "extwr", extwr, int(extwr*1000/NR)/10
10 -print "sexbr", sexbr, int(sexbr*1000/NR)/10
-print "sexwr", sexwr, int(sexwr*1000/NR)/10

# print "movi", movi, "movi64", movi64, "movi128", movi128
# print "addi", addi, "addi32", addi32, "addi64", addi64, "addi128", addi128
15 # print "subi", subi, "subi32", subi32, "subi64", subi64, "subi128", subi128
-}
+
#function p(a, b){
20 # print "a", b, int(b*100/NR)
#}

#!/(j+jl|b|bl)(ge|gt|le|lt|ne|eq|pl|mi|hi|hs|lo|ls)?\.d/ {
# stored = $0
# sub(/\.d/, "", stored)
25 # getline
# print $0
# print stored
# nexte()
#}
30 #
{ print $0 }
```